

LA-UR-22-20924

Approved for public release; distribution is unlimited.

Title: BEE-FY22-1: Enhanced BEE Client 2.3.6.01 - LANL ATDM ST/STNS01-44
Milestone Completion Documentation

Author(s): Randles, Timothy C.

Intended for: ECP Milestone documentation

Issued: 2022-02-02



Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

BEE-FY22-1: Enhanced BEE Client

2.3.6.01 – LANL ATDM ST / STNS01-44 Milestone Completion Documentation

Tim Randles
for the BEE Team

January 31, 2022

LA-UR-22-

Activity Description

Develop a new graphical BEE client for managing workflows.



Execution Plan

To complete this activity the BEE team will investigate a number of existing open source Common Workflow Language-compliant graphical applications. These applications will be evaluated to see if they are good candidates to be extended to interface with the BEE workflow system.

If no existing application can be easily adapted to work with BEE, the team will write a custom application.



Completion Criteria

When this activity is complete:

1. BEE will have a new graphical client
2. the graphical client will support the existing BEE command line client functionality
 1. submit a workflow
 2. list workflows
 3. start a workflow
 4. query a workflow
 5. pause a workflow
 6. resume a workflow
 7. cancel a workflow
 8. copy a workflow
 9. re-execute a workflow
3. the graphical client will allow the user to visualize the current state of their workflow as a DAG



Evaluation of existing graphical applications

The team's evaluation of existing CWL applications quickly narrowed to an examination of Rabix Composer (<https://github.com/rabix/composer>). While Rabix Composer seems to be the ideal tool, the developers have stated that it is no longer supported and has not been updated for newer versions of CWL. The BEE team felt that it was too much burden to fork and maintain a large code that the original developers have stopped supporting.

The team also considered extending Sandia Analysis Workbench (SAW; <https://www.sandia.gov/saw/>). SAW, while a robust and maintained application, contains a huge amount of functionality unrelated to BEE. It was determined that it may be beneficial to eventually extend SAW with the ability to work with BEE. However, relying on SAW as the primary BEE graphical application is not appropriate.

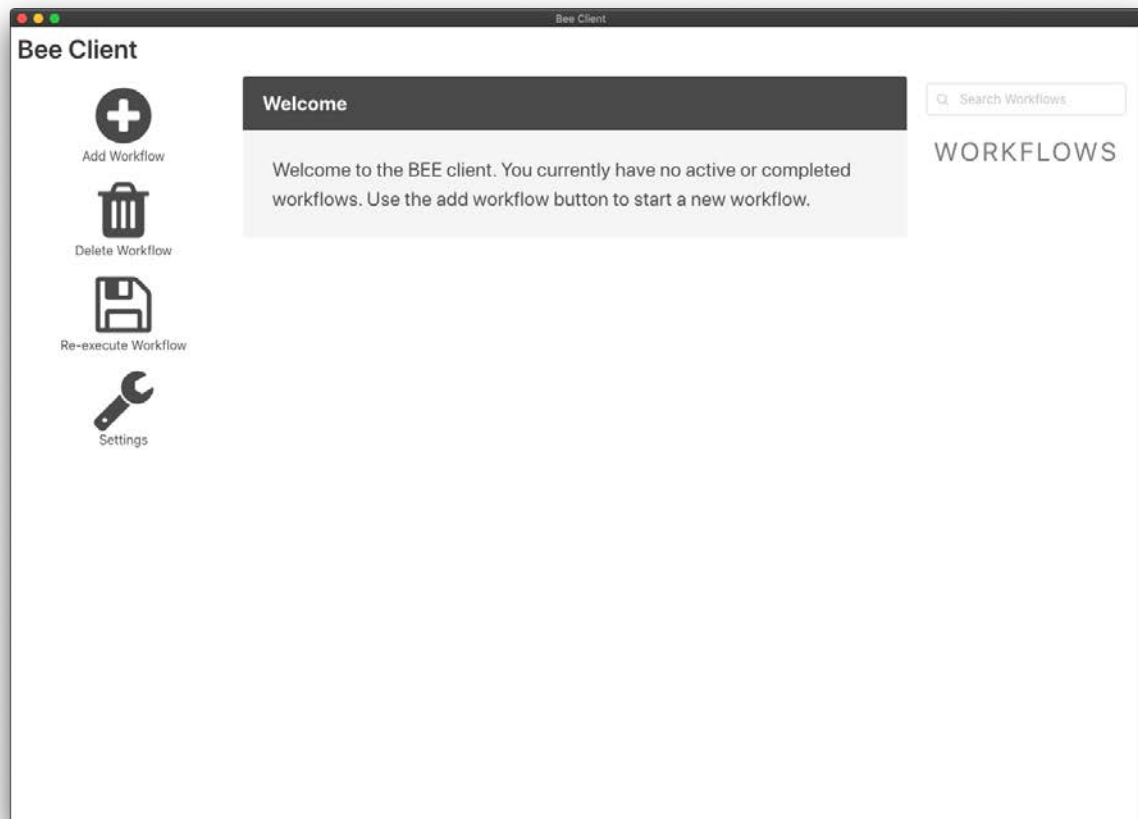
The solution the team implemented was to develop a new Electron-based application. Electron is a cross-platform desktop application development framework that uses Node.js and Chromium, so applications are built using JavaScript, HTML, and CSS. Applications such as Microsoft Teams, Visual Studio Code, and WhatsApp are built using Electron. Electron is open source under the MIT License making it compatible with BEE's Apache 2.0 open source license.



Completion Criteria



1. BEE will have a new graphical client

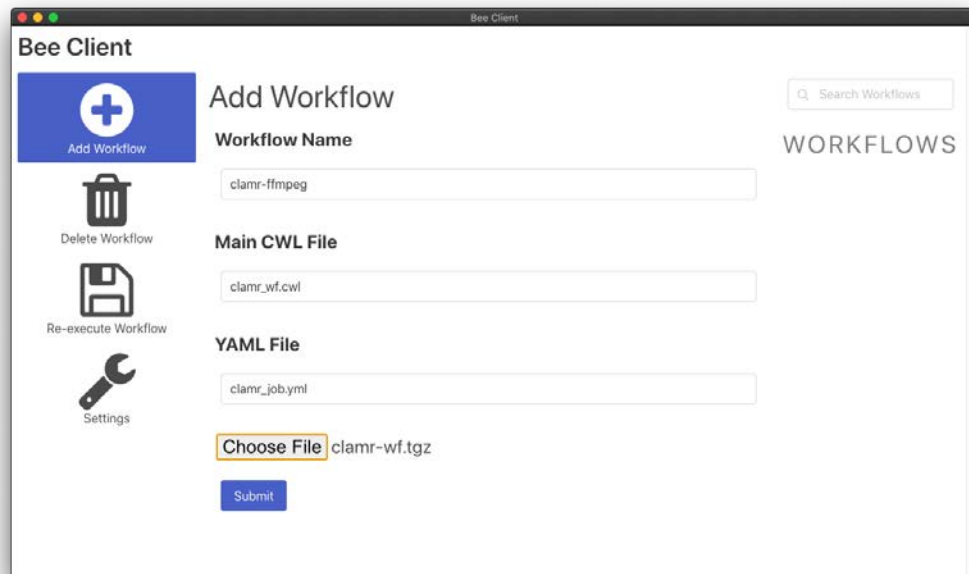


2. the graphical client will support the existing BEE command line client functionality

submit a workflow

A user clicks on the Add Workflow button, fills in the required information, and clicks submit.

The client submits the workflow to the workflow manager, which parses, creates a unique workflow ID, and returns the ID to the client. The client then issues a GET request to retrieve the rest of the workflow information.



The screenshot shows the 'Bee Client' window. On the left is a sidebar with icons for 'Add Workflow' (a plus sign), 'Delete Workflow' (a trash can), 'Re-execute Workflow' (a floppy disk), and 'Settings' (a wrench). The main area is titled 'Add Workflow' and contains three text input fields: 'Workflow Name' (containing 'clamr-ffmpeg'), 'Main CWL File' (containing 'clamr_wf.cwl'), and 'YAML File' (containing 'clamr_job.yml'). Below these is a 'Choose File' button next to the text 'clamr-wf.tgz'. At the bottom is a blue 'Submit' button. In the top right corner, there is a search bar labeled 'Search Workflows' and a section header 'WORKFLOWS'.

```
INFO: 127.0.0.1 - - [02/Feb/2022 14:56:03] "POST /bee_wfm/v1/jobs/ HTTP/1.1" 201 -
INFO: 127.0.0.1 - - [02/Feb/2022 14:56:05] "GET /bee_wfm/v1/jobs/623975d4-8acb-4141-b759-043e142b7088?wf_id=623975d4-8acb-4141-b759-043e142b7088 HTTP/1.1" 200 -
```

workflow manager log entry

2/2/22



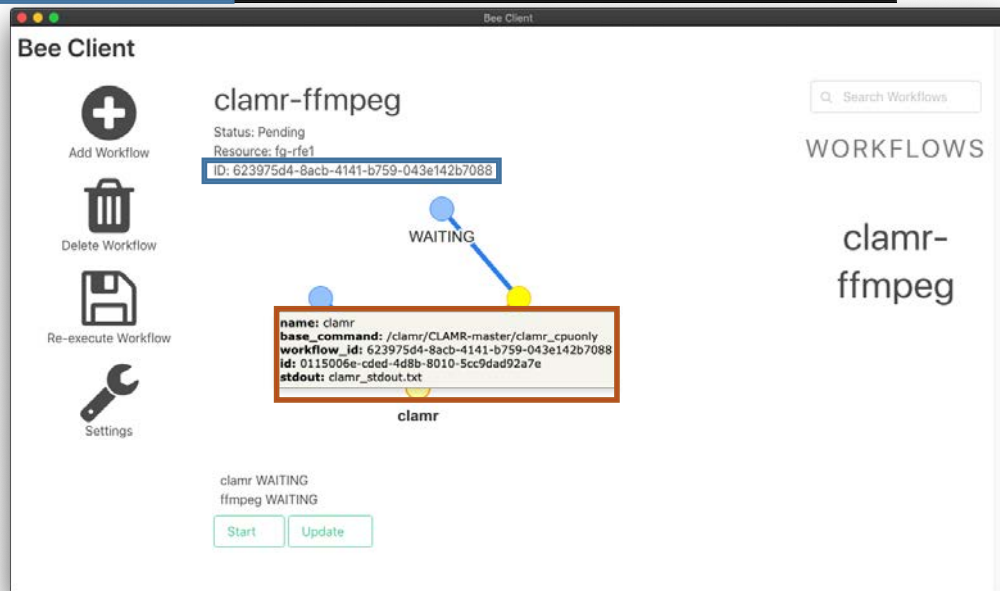
2. the graphical client will support the existing BEE command line client functionality

submit a workflow

```
> BEEClient@1.0.0 start
> electron .

{
  "msg": "Workflow uploaded",
  "status": "ok",
  "tasks": [
    {
      "id": "0115006e-cded-4d8b-8010-5cc9dad92a7e",
      "command": "/clamr/CLAMR-master/clamr_cpuonly",
      "name": "clamr",
      "status": "WAITING",
      "wf_id": "623975d4-8acb-4141-b759-043e142b7088"
    },
    {
      "command": "ffmpeg -y",
      "name": "ffmpeg",
      "status": "WAITING",
      "wf_id": "623975d4-8acb-4141-b759-043e142b7088"
    }
  ]
}
```

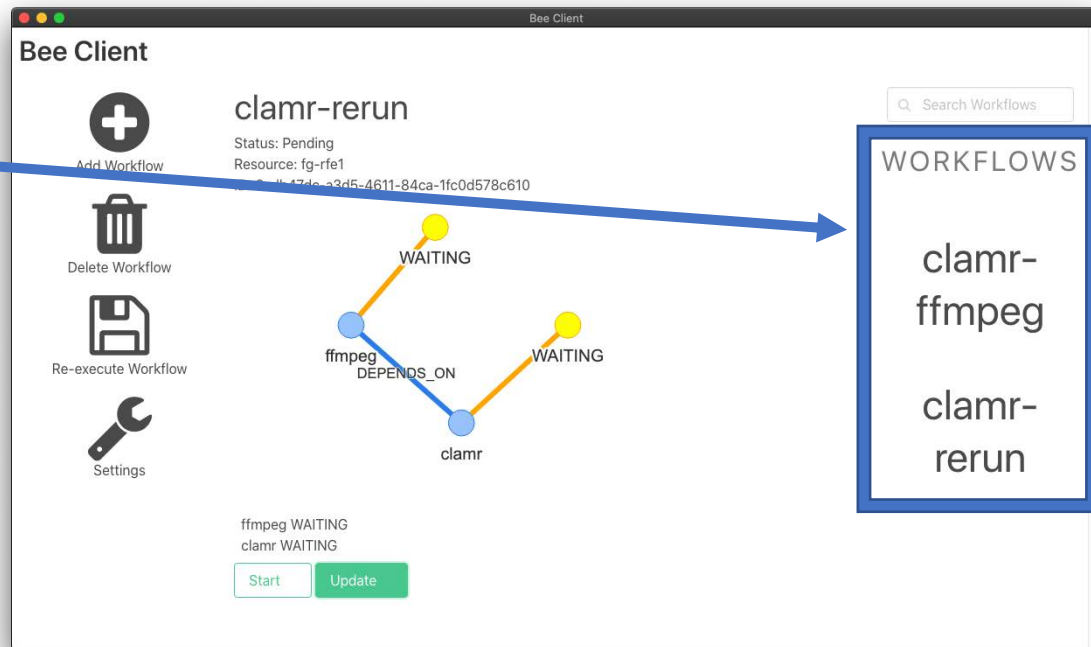
The client GUI updates to show submitted the workflow. The workflow manager parses the workflow and returns information, such as the DAG and the workflow and task IDs back to the GUI client. Some of the IDs are highlighted here.



2. the graphical client will support the existing BEE command line client functionality

list workflows

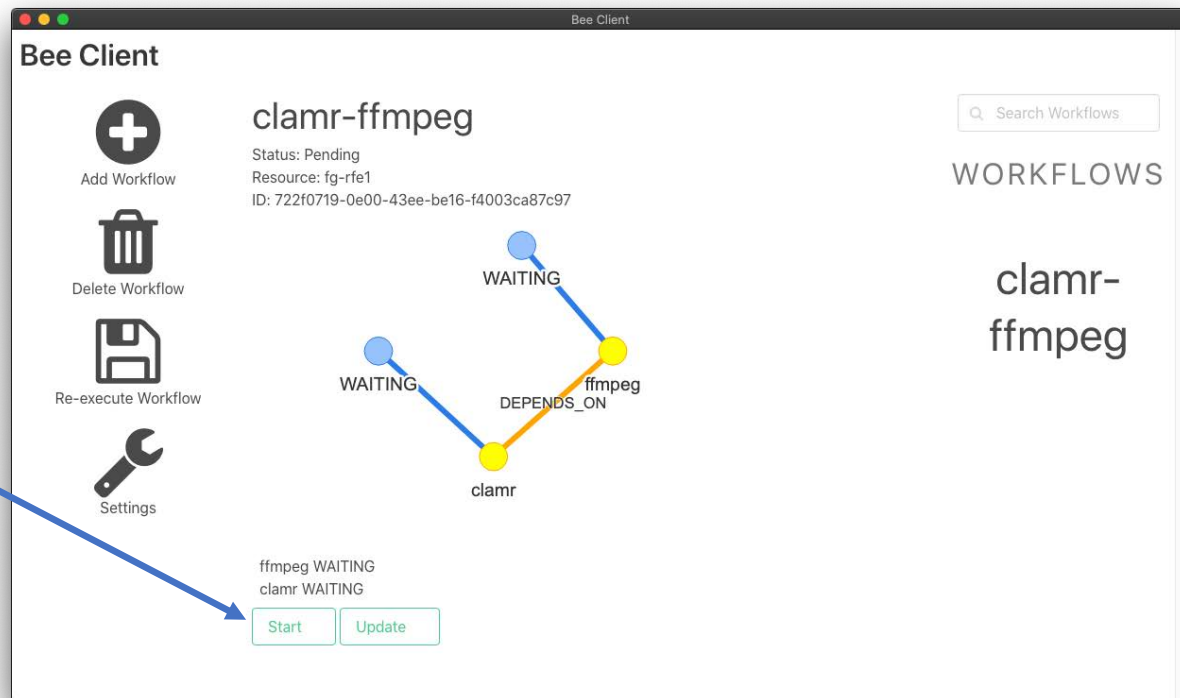
The client presents a list of available workflows. Here we see the clamr-ffmpeg workflow listed as well as a re-execution of the original workflow.



2. the graphical client will support the existing BEE command line client functionality

start a workflow

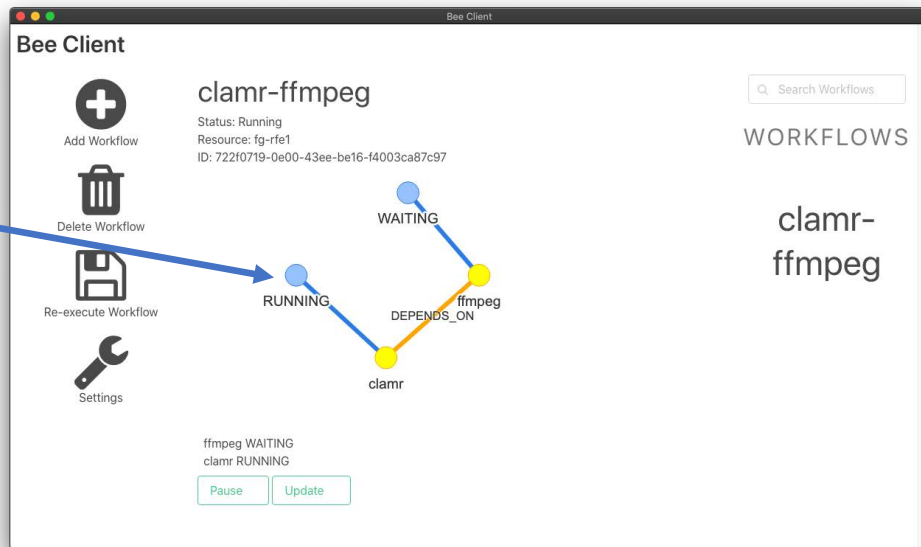
The user presses the “Start” button to begin the execution of a workflow.



2. the graphical client will support the existing BEE command line client functionality

start a workflow

The first task's state changes to "RUNNING" and a job is executing on the cluster.



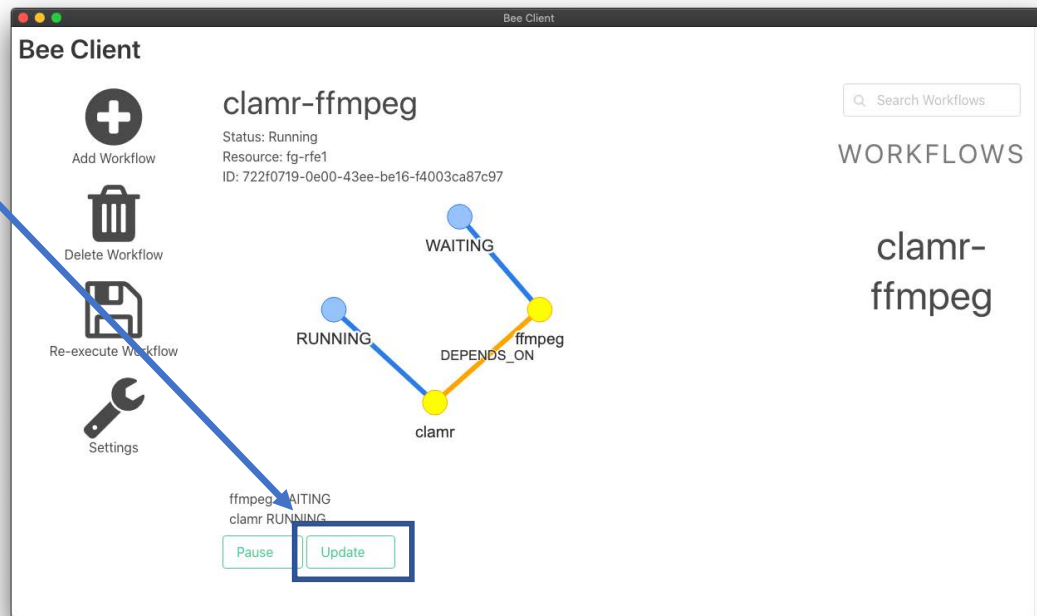
```
[trandles@fg-rfe1 archives]$ squeue
      JOBID PARTITION   NAME     USER ST       TIME  NODES NODELIST(REASON)
    433046  standard clamr-af trandles R        0:45      1 fg014
```



2. the graphical client will support the existing BEE command line client functionality

query a workflow

A user can click the “Update” button at any time and the client will query the workflow manager to get the state of a workflow.



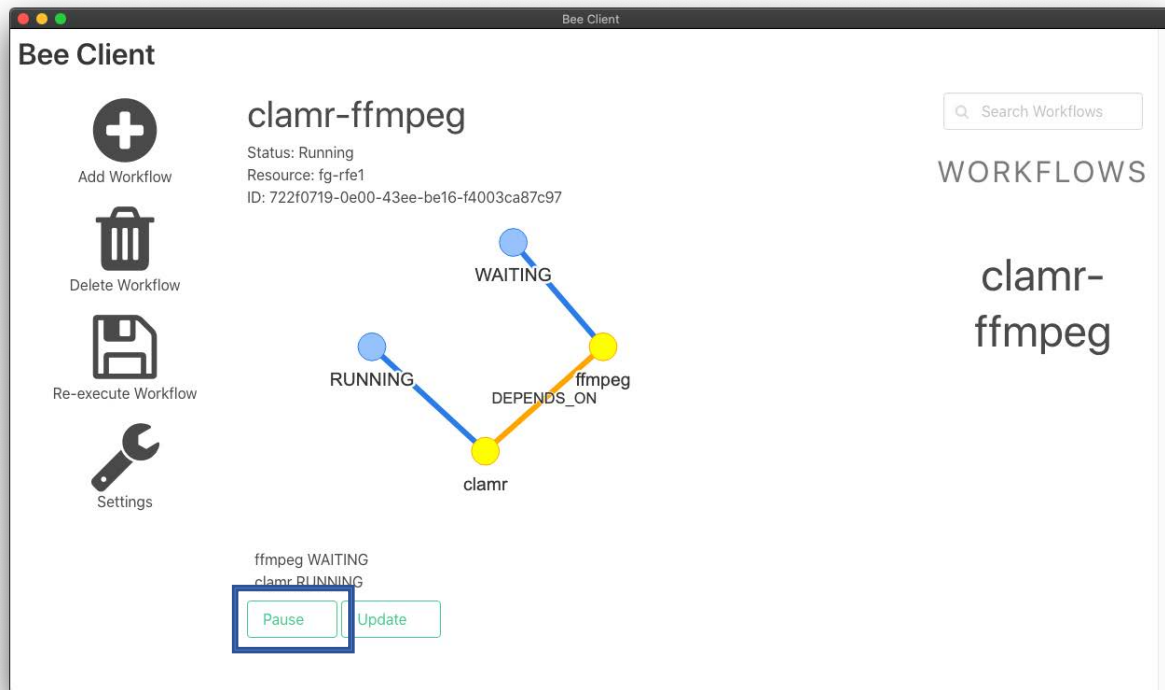
```
INFO: 127.0.0.1 - - [02/Feb/2022 14:40:37] "GET /bee_wfm/v1/jobs/722f0719-0e00-43ee-be16-f4003ca87c97?wf_id=722f0719-0e00-43ee-be16-f4003ca87c97 HTTP/1.1" 200 -
INFO: 127.0.0.1 - - [02/Feb/2022 14:40:42] "GET /bee_wfm/v1/jobs/722f0719-0e00-43ee-be16-f4003ca87c97?wf_id=722f0719-0e00-43ee-be16-f4003ca87c97 HTTP/1.1" 200 -
INFO: 127.0.0.1 - - [02/Feb/2022 14:40:55] "GET /bee_wfm/v1/jobs/722f0719-0e00-43ee-be16-f4003ca87c97?wf_id=722f0719-0e00-43ee-be16-f4003ca87c97 HTTP/1.1" 200 -
```



2. the graphical client will support the existing BEE command line client functionality

pause a workflow

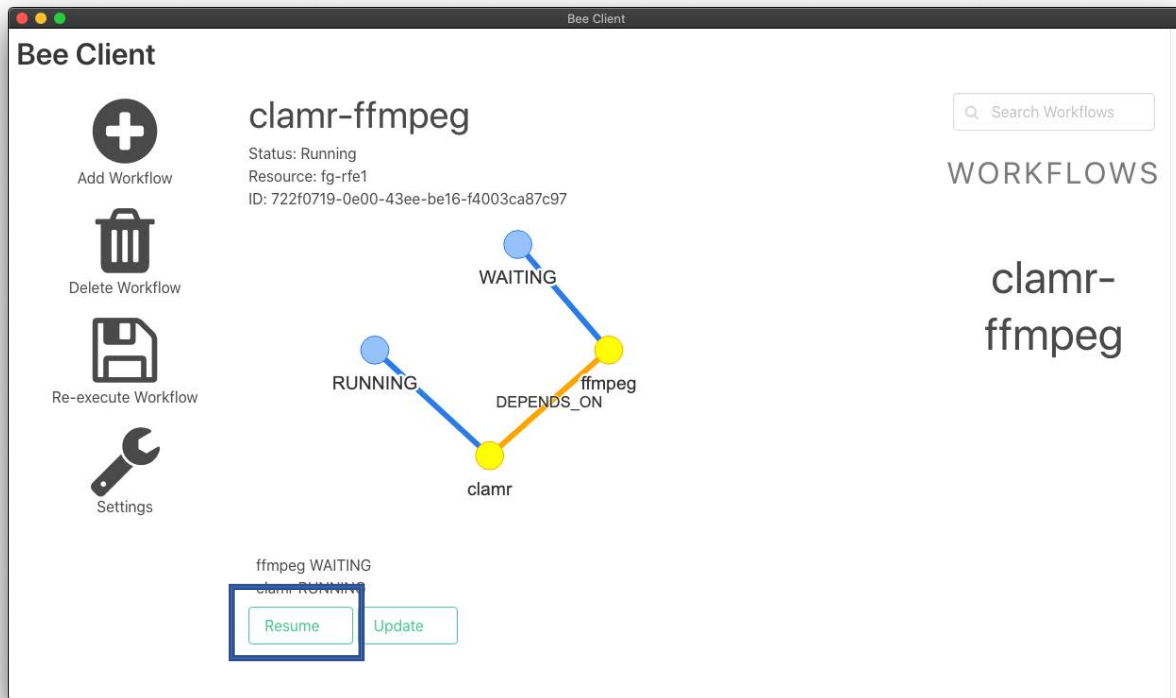
When a workflow is running the Submit button turns into a Pause button. When Pause is pressed any running tasks will continue to completion but no new tasks will be submitted.



2. the graphical client will support the existing BEE command line client functionality

resume a workflow

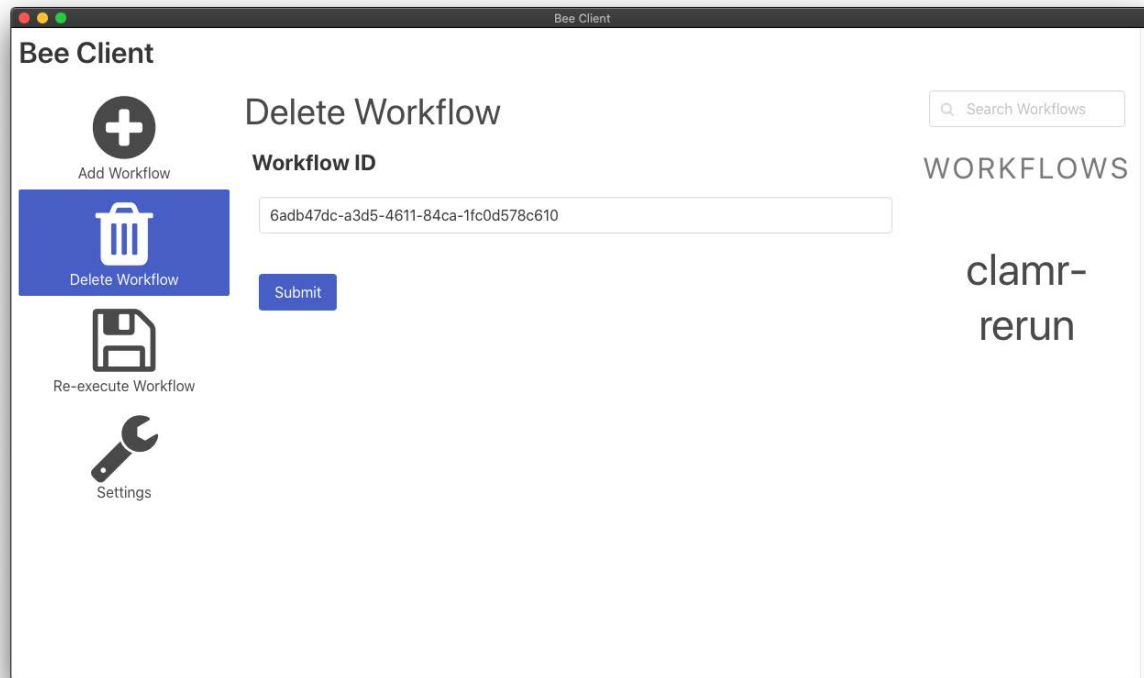
When a workflow is paused a resume button appears. Resume will continue running the workflow from the next READY task.



2. the graphical client will support the existing BEE command line client functionality

cancel a workflow

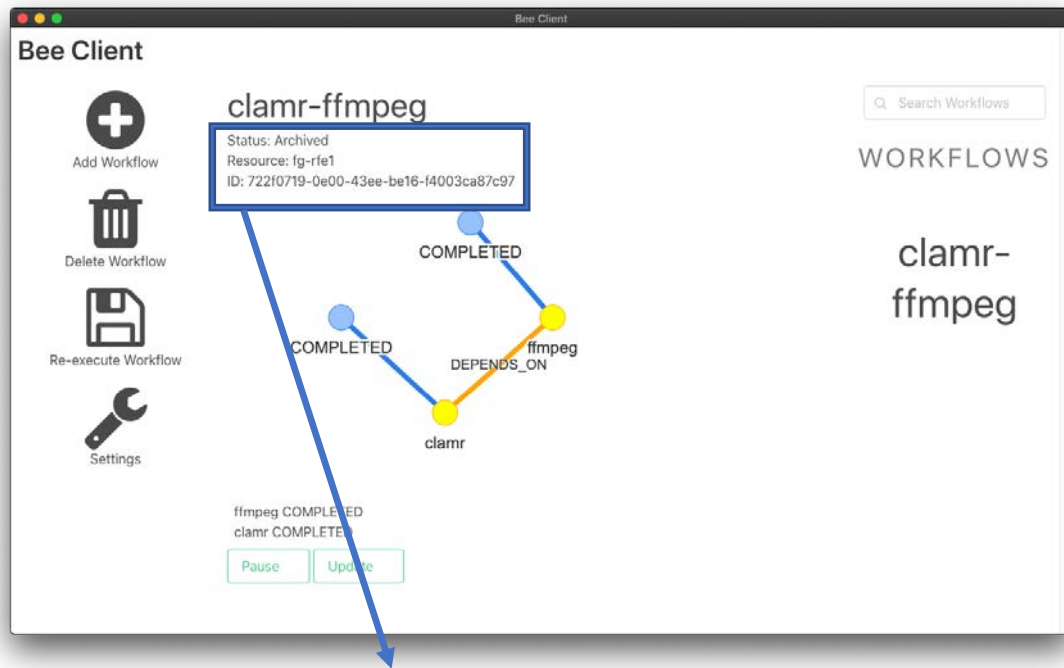
To cancel a running workflow using the graphical client a user deletes the workflow. This kills any running tasks and removes the workflow from the workflow manager.



2. the graphical client will support the existing BEE command line client functionality

copy a workflow

By default, BEE will create an archive of your workflow when it completes. This can be considered a copy, which may be shared and re-executed by the other users.



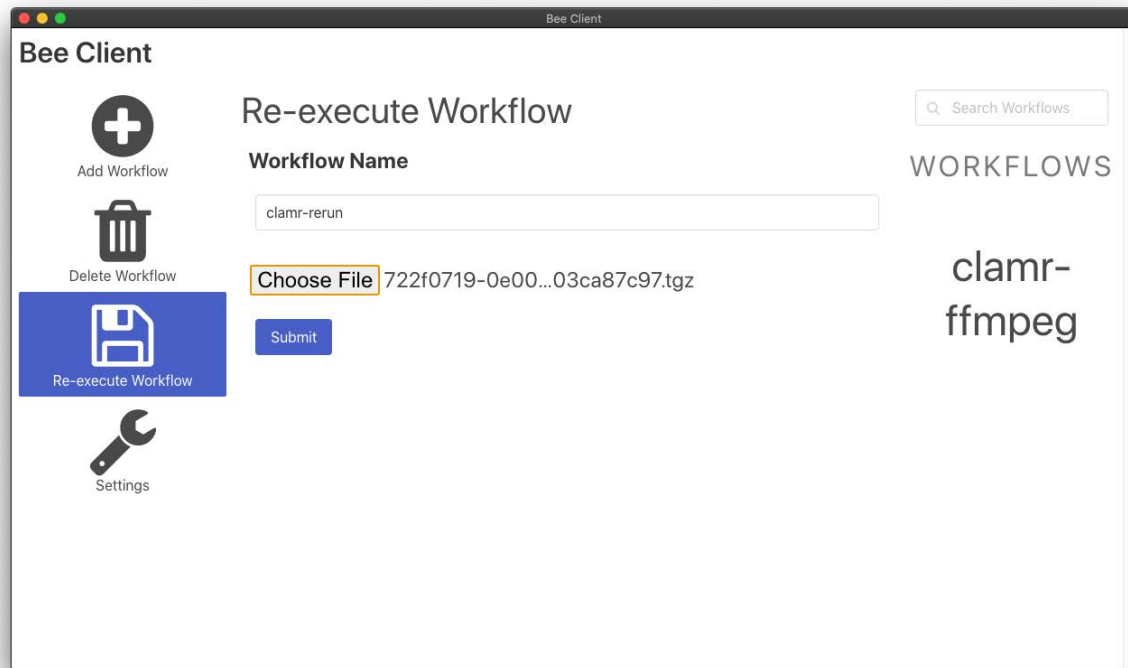
```
[trandles@fg-rfe1 ~]$ ls -l .beeflow/archives/722f0719-0e00-43ee-be16-f4003ca87c97.tgz
-rw-rw---- 1 trandles trandles 42064 Feb  2 14:41 .beeflow/archives/722f0719-0e00-43ee-be16-f4003ca87c97.tgz
```



2. the graphical client will support the existing BEE command line client functionality

re-execute a workflow

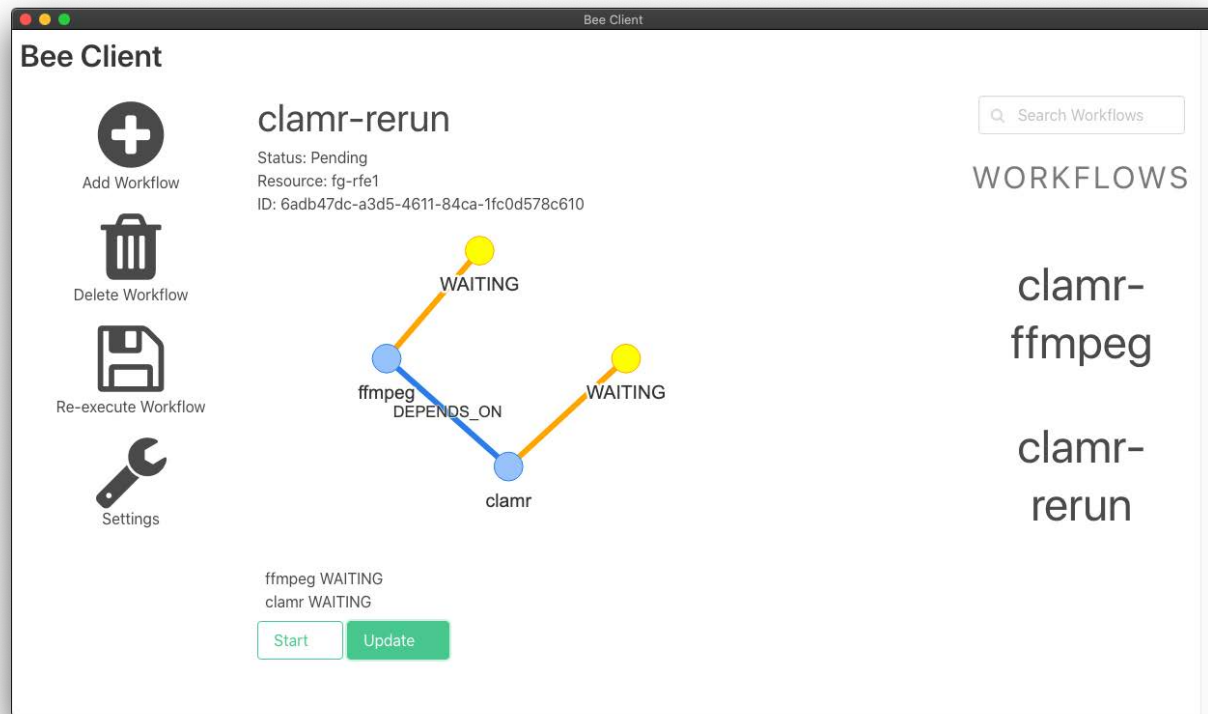
To re-execute a workflow, a user gives the workflow a new name and selects the archive file of a previously-executed workflow.



2. the graphical client will support the existing BEE command line client functionality

re-execute a workflow

The workflow manager reprocesses the archived workflow, resetting the task states and giving the workflow a new unique ID. Once this is finished the workflow may be started.



3. the graphical client will allow the user to visualize the current state of their workflow as a DAG

The BEE client presents the user with a graphical representation of their workflow DAG. Dependencies are indicated using the `DEPENDS_ON` tag. In this picture task nodes are colored yellow. The blue metadata nodes indicate the task's state.

